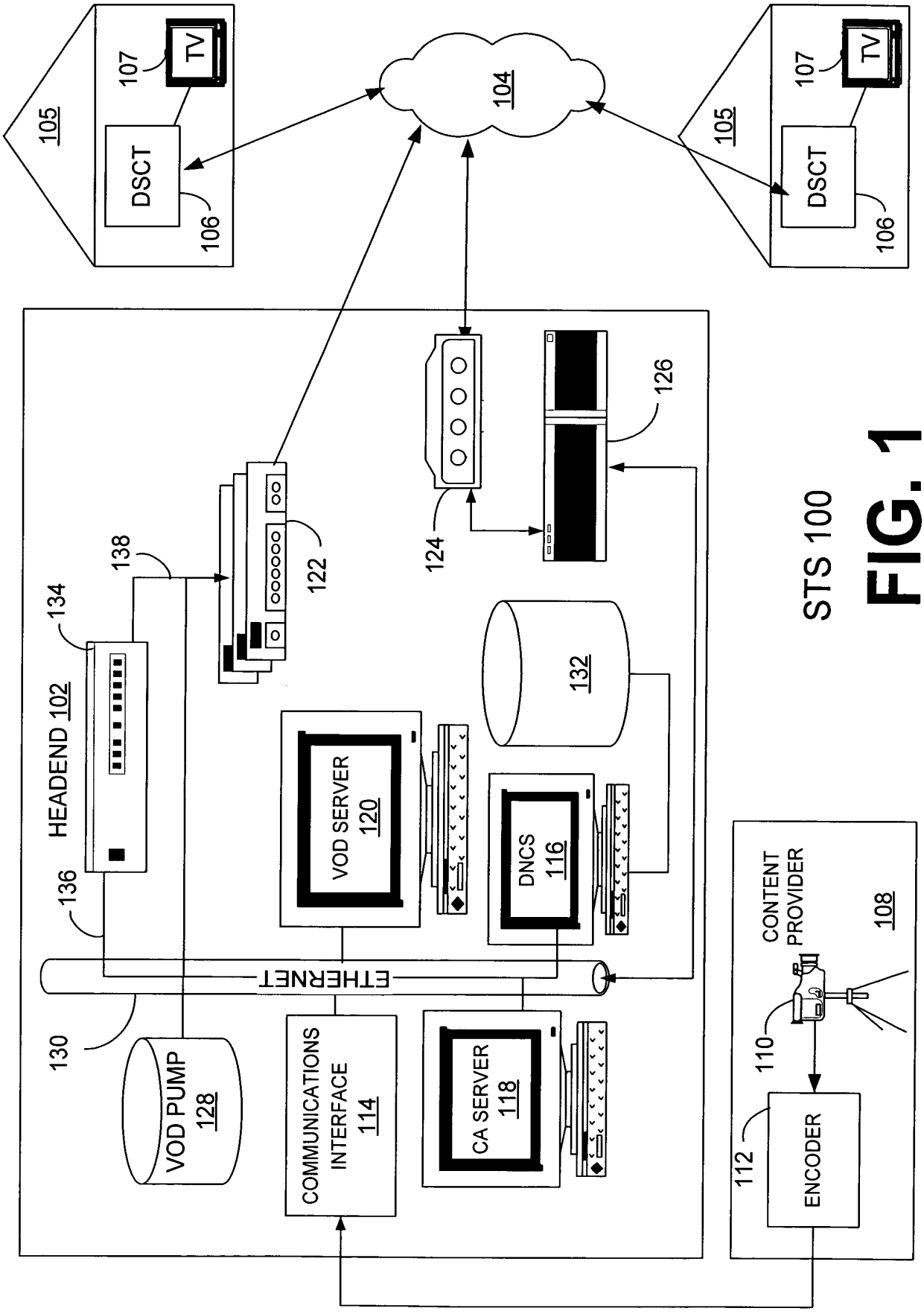
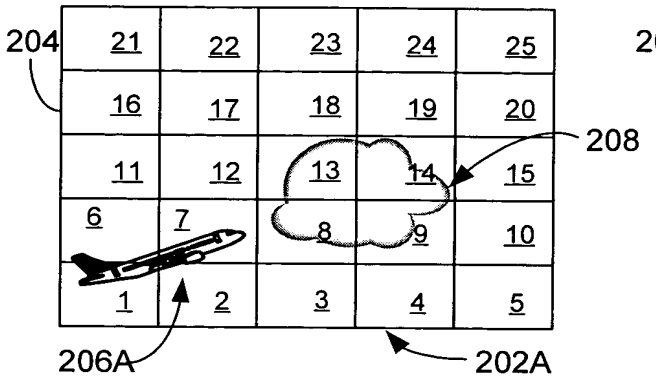
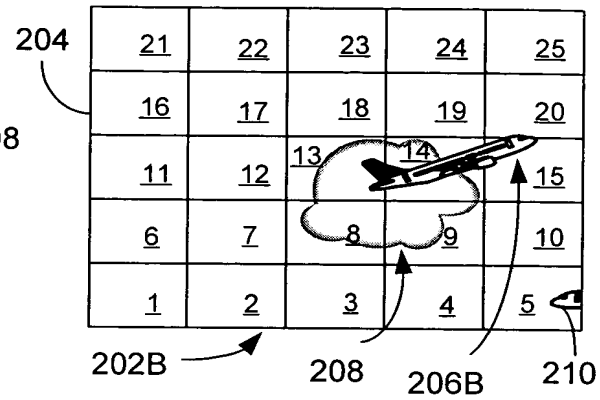

105
107
TV
DSCT
106
104
105
107
TV
DSCT
106
126
124
122
138
134
HEADEND 102
136
132
VOD SERVER 120
DNCS 116
ETHERNET
130
VOD PUMP 128
COMMUNICATIONS INTERFACE 114
CA SERVER 118
CONTENT PROVIDER
108
110
112
ENCODER
STS 100

FIG. 1



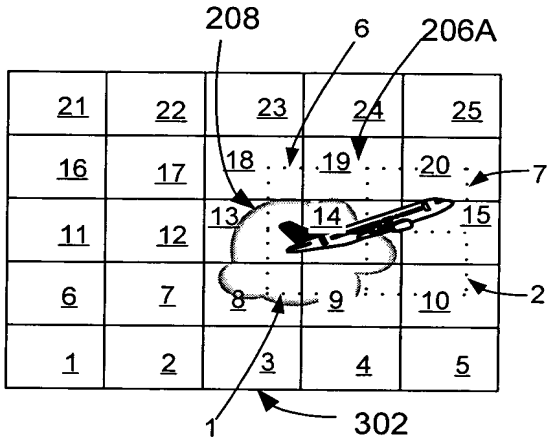
PRIOR ART

FIG. 2A



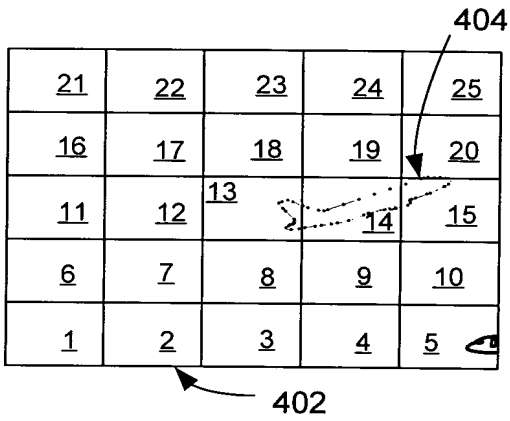
PRIOR ART

FIG. 2B



PRIOR ART

FIG. 3



PRIOR ART

FIG. 4

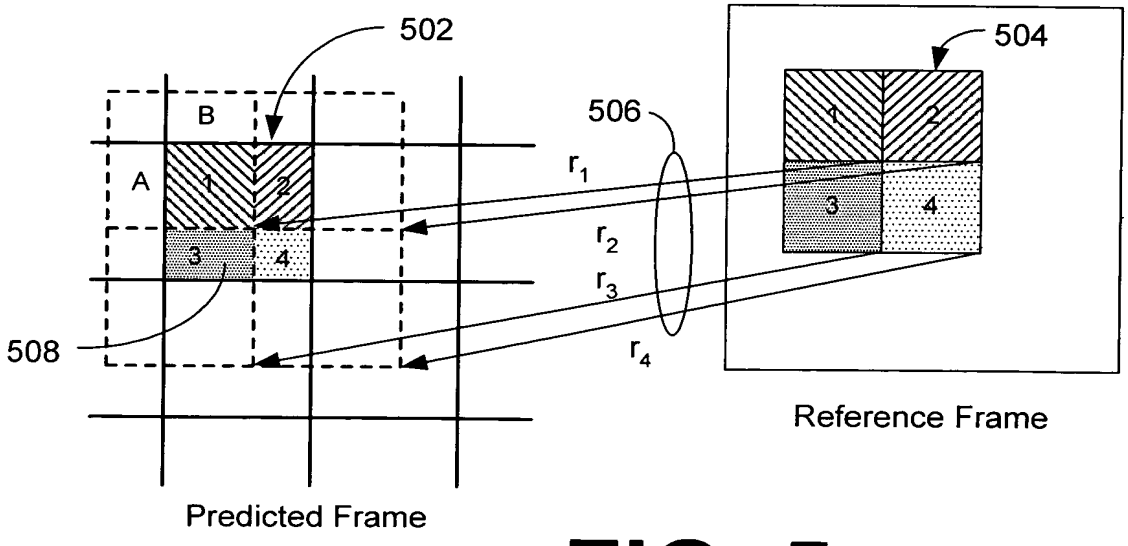
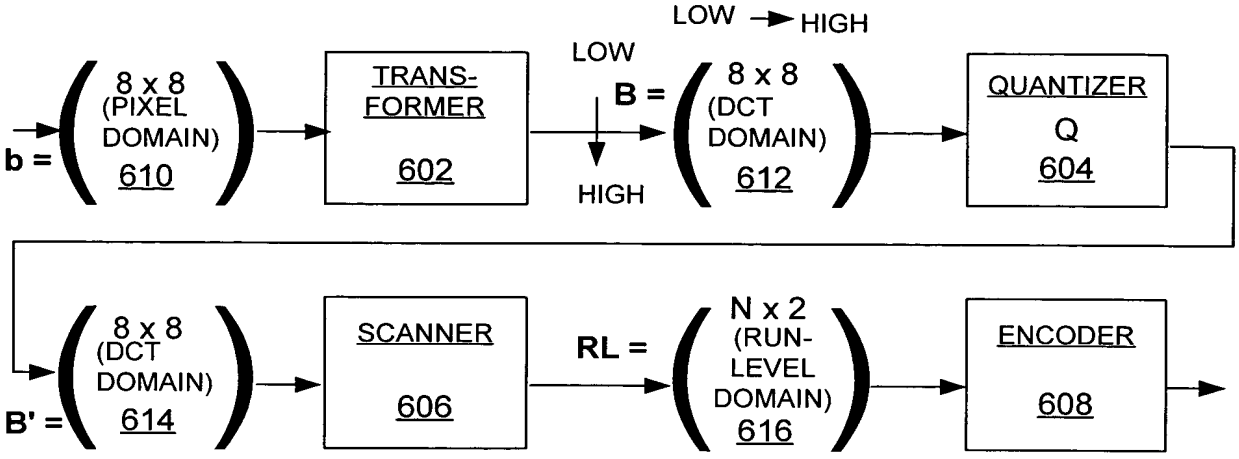
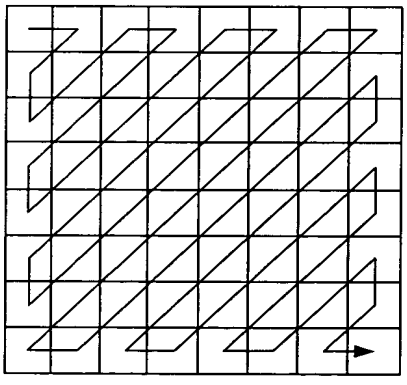


FIG. 5



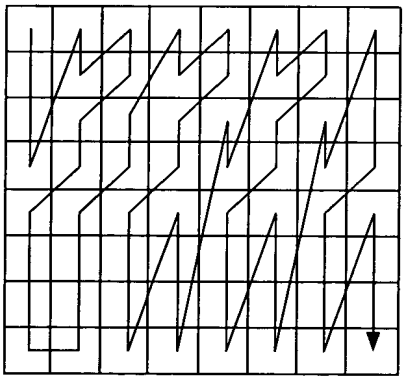
PRIOR ART

FIG. 6



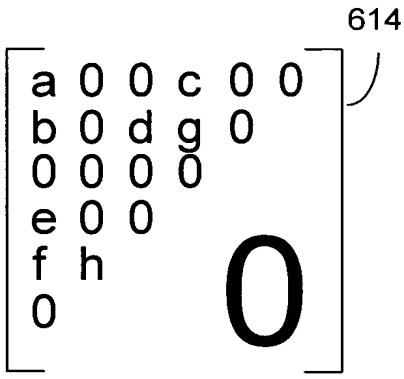
PRIOR ART

FIG. 7A



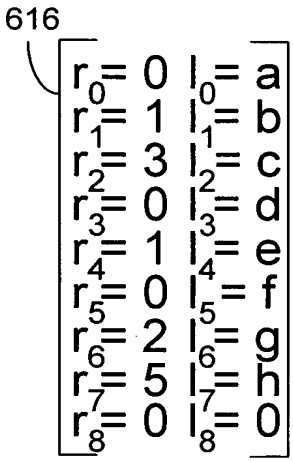
PRIOR ART

FIG. 7B



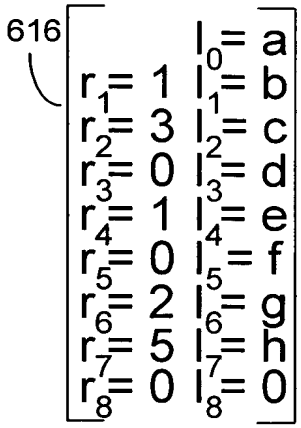
PRIOR ART

FIG. 8A



PRIOR ART

FIG. 8B



PRIOR ART

FIG. 8C

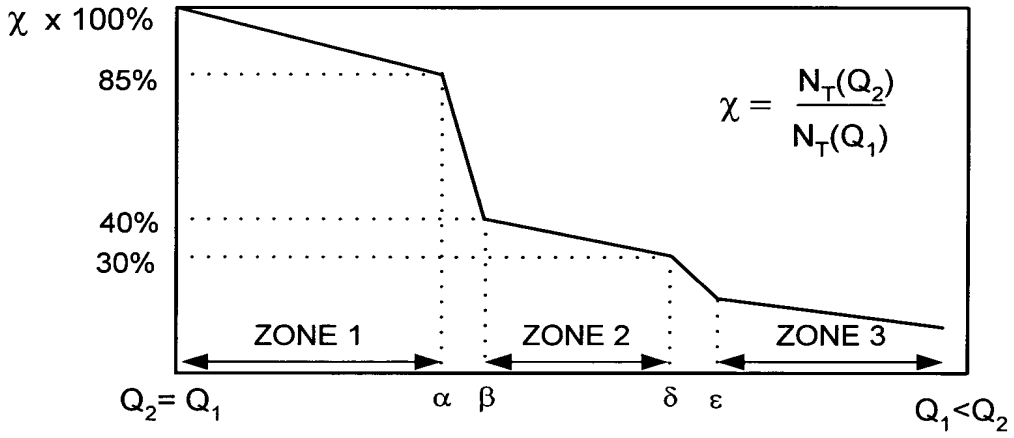


FIG. 10

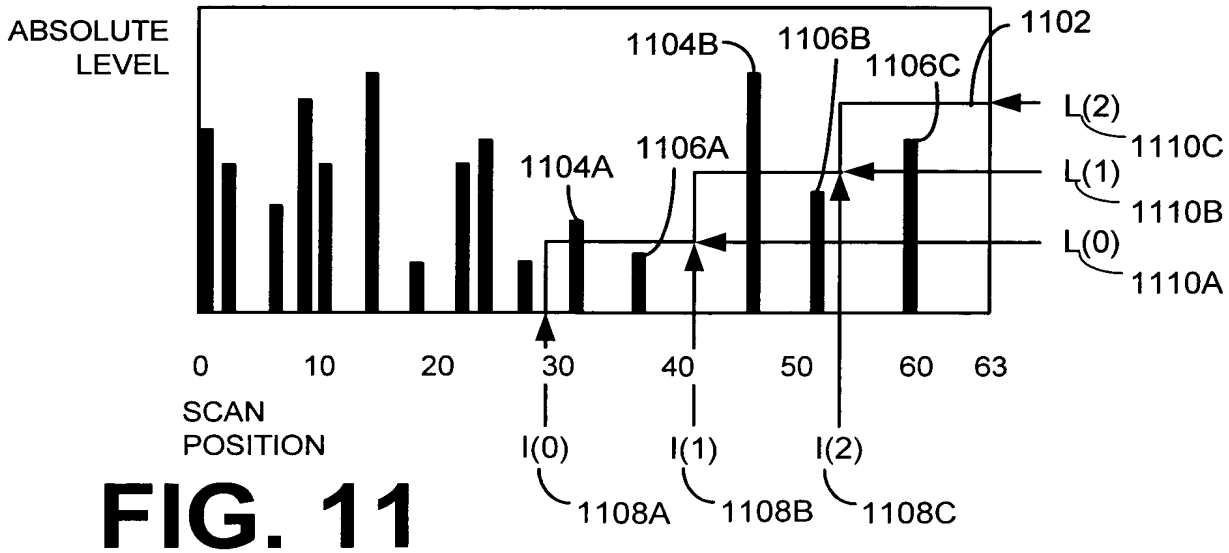


FIG. 11

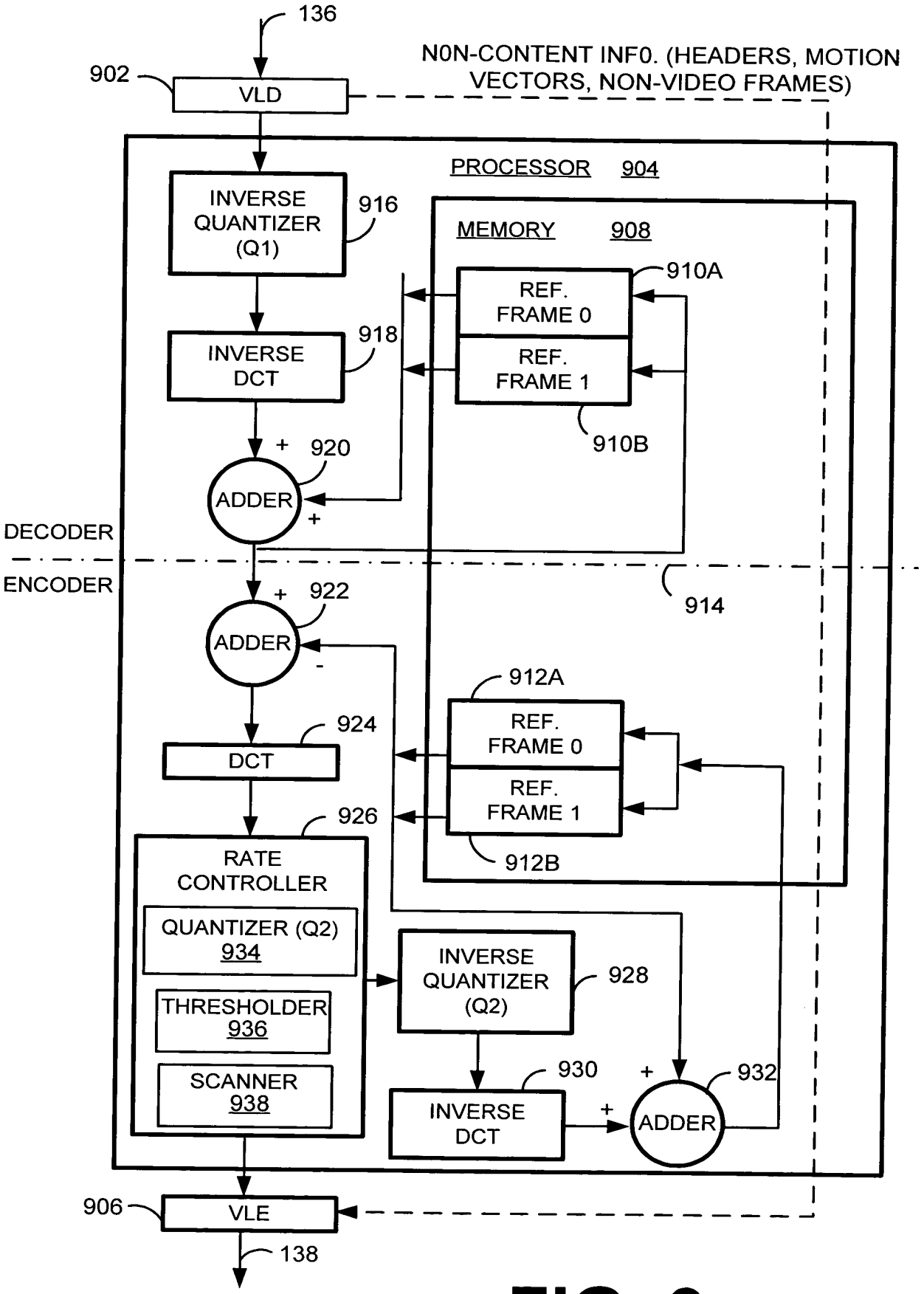

136
902
VLD
NON-CONTENT INFO. (HEADERS, MOTION VECTORS, NON-VIDEO FRAMES)
PROCESSOR 904
INVERSE QUANTIZER (Q1)
916
MEMORY 908
910A
REF. FRAME 0
REF. FRAME 1
910B
INVERSE DCT
918
+
920
ADDER
+
DECODER
ENCODER
914
+
922
ADDER
-
912A
REF. FRAME 0
REF. FRAME 1
912B
924
DCT
926
RATE CONTROLLER
QUANTIZER (Q2) 934
THRESHOLDER 936
SCANNER 938
INVERSE QUANTIZER (Q2)
928
930
INVERSE DCT
+
+
932
ADDER
906
VLE
138

FIG. 9

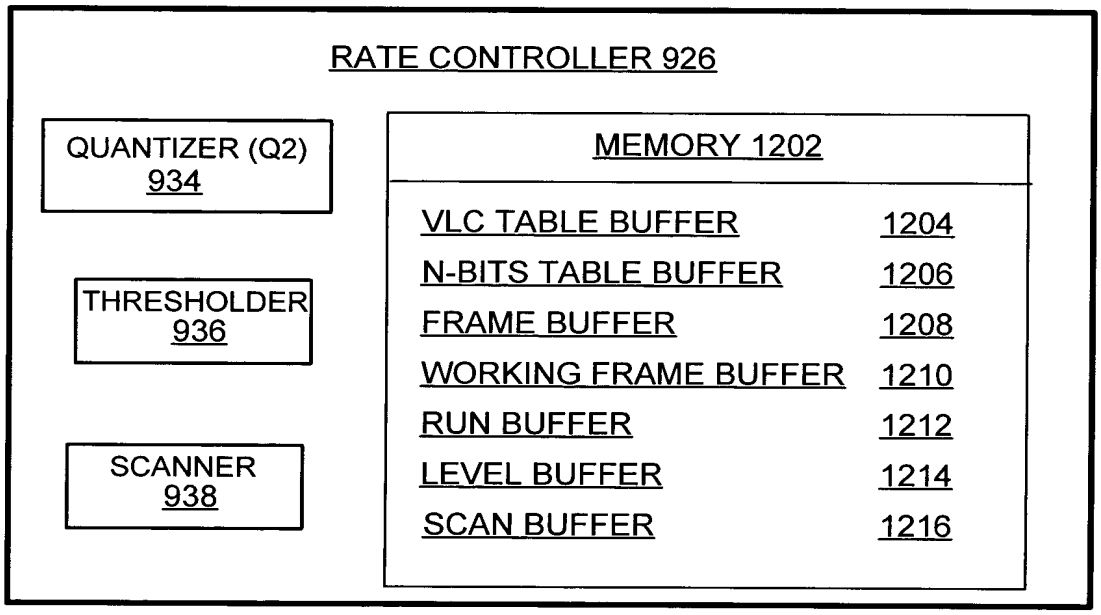

RATE CONTROLLER 926
QUANTIZER (Q2) 934
THRESHOLDER 936
SCANNER 938
MEMORY 1202
VLC TABLE BUFFER 1204
N-BITS TABLE BUFFER 1206
FRAME BUFFER 1208
WORKING FRAME BUFFER 1210
RUN BUFFER 1212
LEVEL BUFFER 1214
SCAN BUFFER 1216

FIG. 12

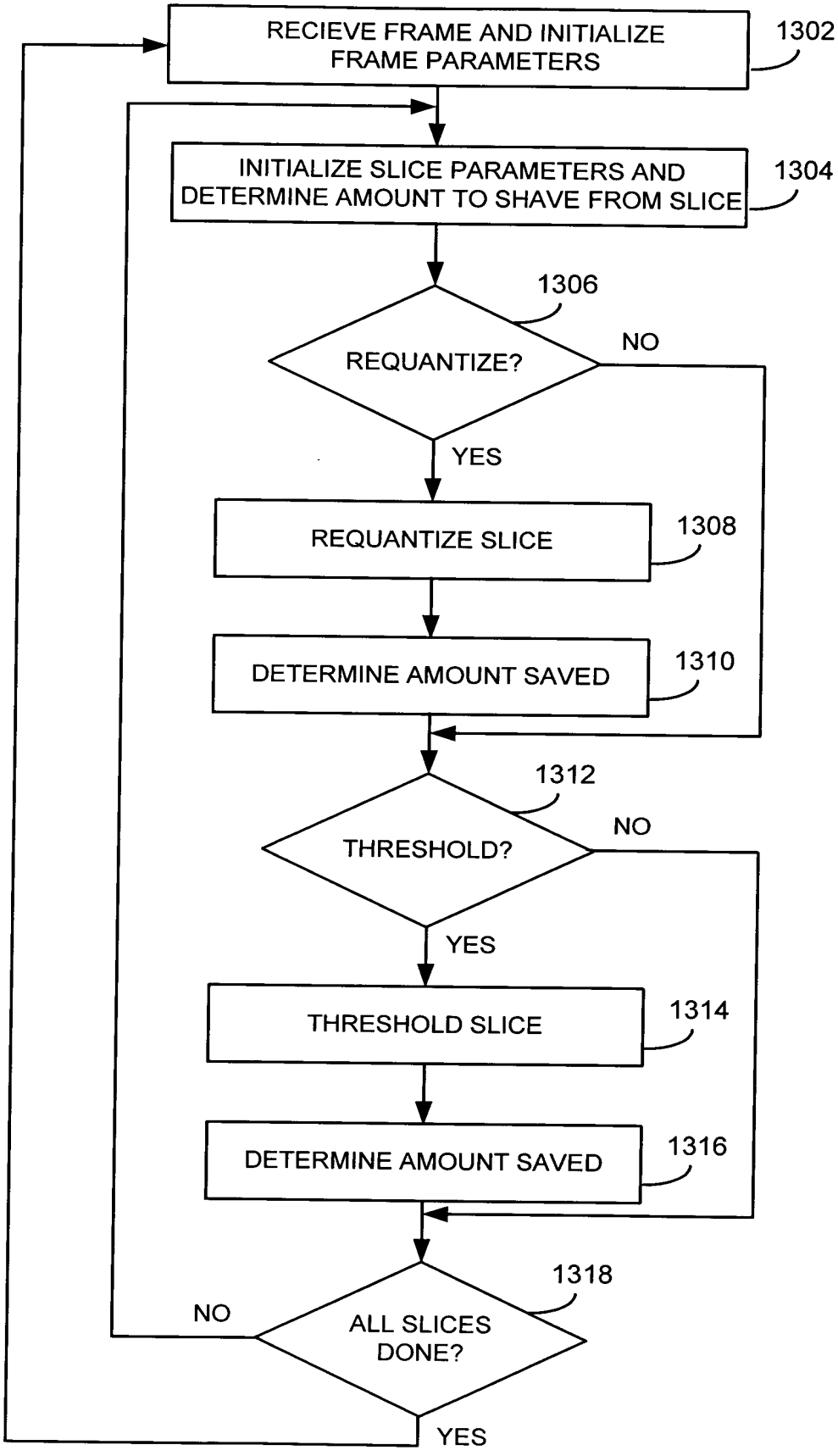

RECIEVE FRAME AND INITIALIZE FRAME PARAMETERS
1302
INITIALIZE SLICE PARAMETERS AND DETERMINE AMOUNT TO SHAVE FROM SLICE
1304
1306
REQUANTIZE?
NO
YES
REQUANTIZE SLICE
1308
DETERMINE AMOUNT SAVED
1310
1312
THRESHOLD?
NO
YES
THRESHOLD SLICE
1314
DETERMINE AMOUNT SAVED
1316
1318
NO
ALL SLICES DONE?
YES

FIG. 13

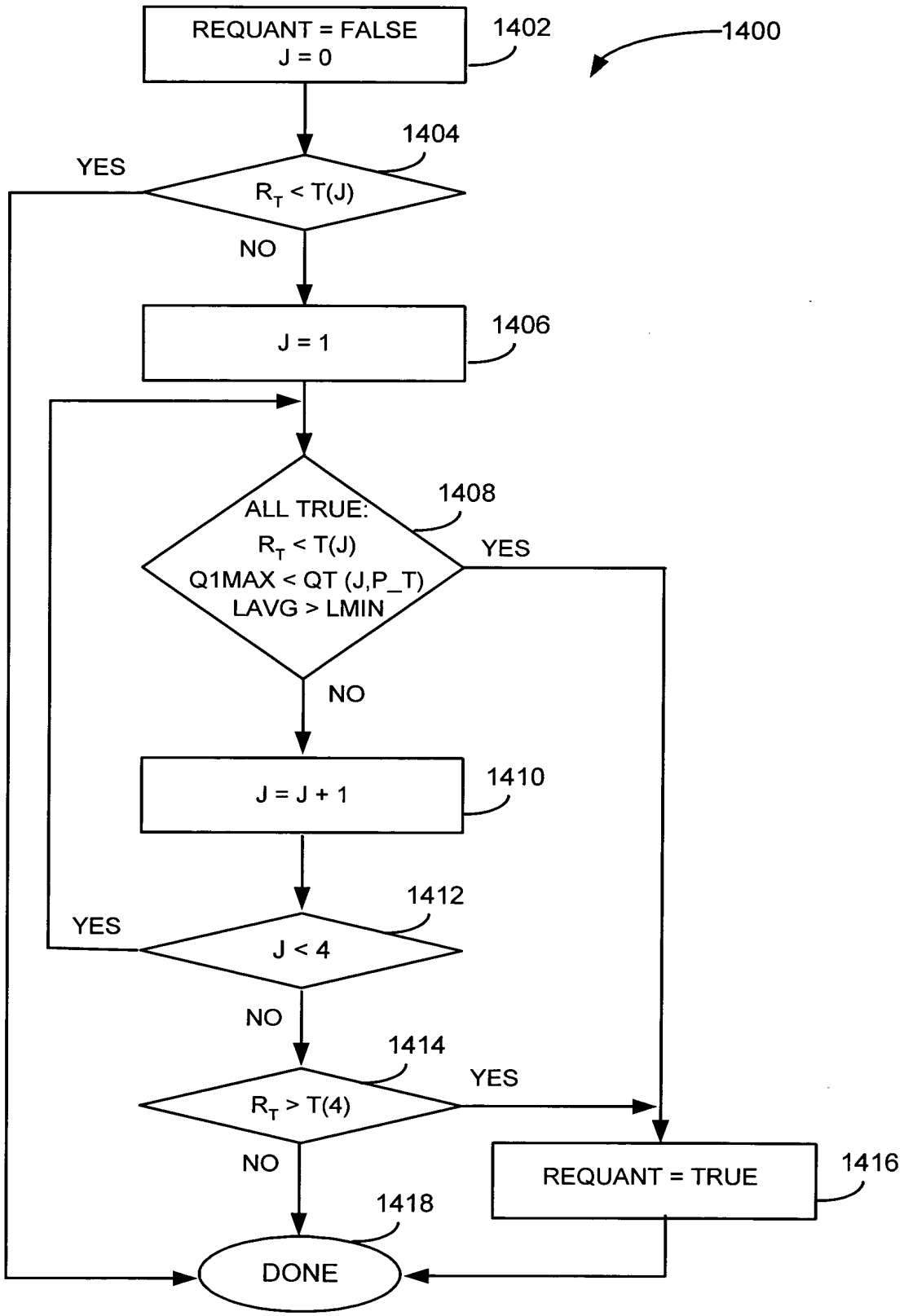

REQUANT = FALSE
J = 0
1402
1400
1404
YES
R_T < T(J)
NO
J = 1
1406
1408
ALL TRUE:
R_T < T(J)
Q1MAX < QT (J,P_T)
LAVG > LMIN
YES
NO
J = J + 1
1410
1412
YES
J < 4
NO
1414
YES
R_T > T(4)
NO
REQUANT = TRUE
1416
1418
DONE

FIG. 14

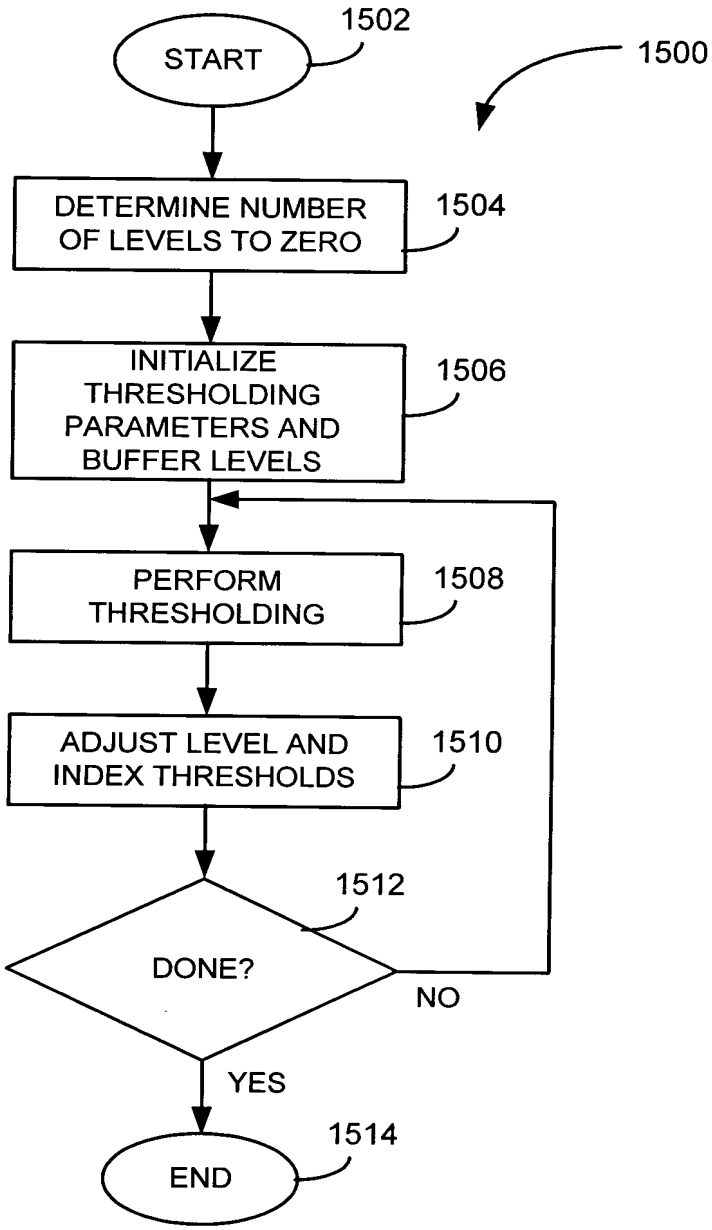

1502
START
1500
DETERMINE NUMBER OF LEVELS TO ZERO
1504
INITIALIZE THRESHOLDING PARAMETERS AND BUFFER LEVELS
1506
PERFORM THRESHOLDING
1508
ADJUST LEVEL AND INDEX THRESHOLDS
1510
1512
DONE?
NO
YES
1514
END

FIG. 15

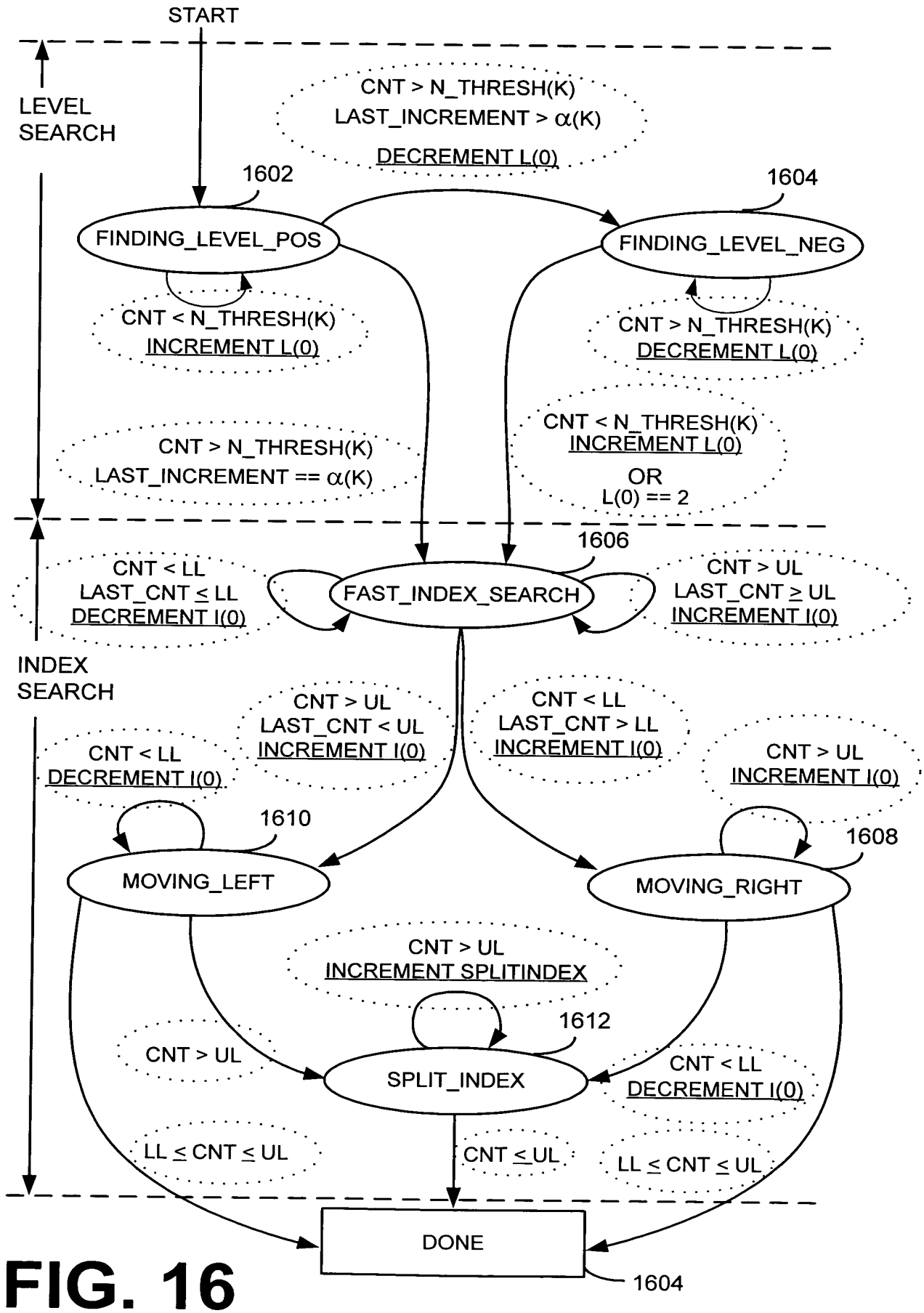

START
LEVEL SEARCH
CNT > N_THRESH(K)
LAST_INCREMENT > α(K)
DECREMENT L(0)
1602
1604
FINDING_LEVEL_POS
FINDING_LEVEL_NEG
CNT < N_THRESH(K)
INCREMENT L(0)
CNT > N_THRESH(K)
DECREMENT L(0)
CNT > N_THRESH(K)
LAST_INCREMENT == α(K)
CNT < N_THRESH(K)
INCREMENT L(0)
OR
L(0) == 2
1606
CNT < LL
LAST_CNT ≤ LL
DECREMENT I(0)
FAST_INDEX_SEARCH
CNT > UL
LAST_CNT ≥ UL
INCREMENT I(0)
INDEX SEARCH
CNT > UL
LAST_CNT < UL
INCREMENT I(0)
CNT < LL
LAST_CNT > LL
INCREMENT I(0)
CNT < LL
DECREMENT I(0)
CNT > UL
INCREMENT I(0)
1610
1608
MOVING_LEFT
MOVING_RIGHT
CNT > UL
INCREMENT SPLITINDEX
1612
CNT > UL
SPLIT_INDEX
CNT < LL
DECREMENT I(0)
LL ≤ CNT ≤ UL
CNT ≤ UL
LL ≤ CNT ≤ UL
DONE
1604

FIG. 16

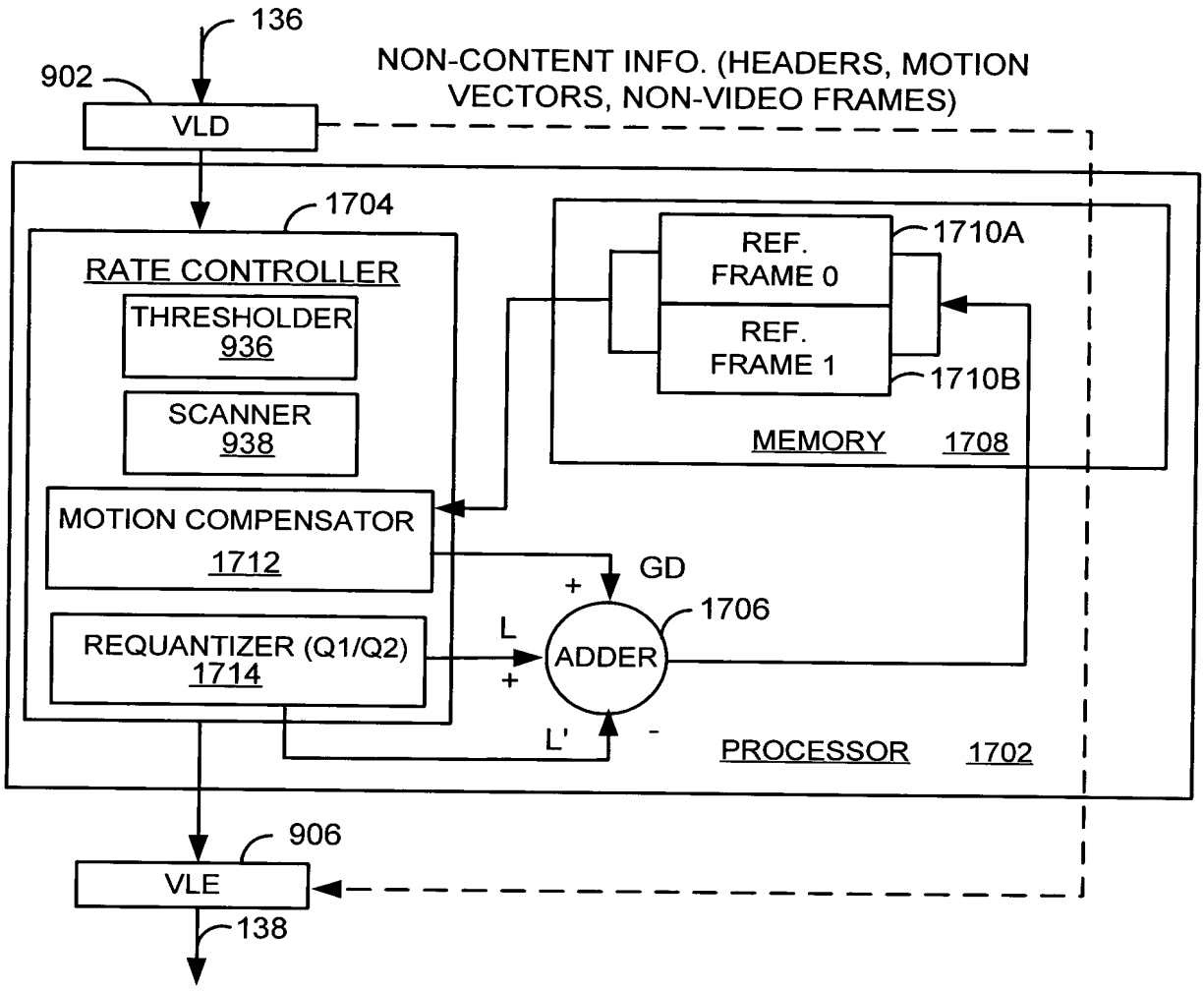

136
902
NON-CONTENT INFO. (HEADERS, MOTION VECTORS, NON-VIDEO FRAMES)
VLD
1704
RATE CONTROLLER
THRESHOLDER 936
SCANNER 938
MOTION COMPENSATOR 1712
REQUANTIZER (Q1/Q2) 1714
REF. FRAME 0
1710A
REF. FRAME 1
1710B
MEMORY 1708
GD
+
L
+
ADDER
1706
L'
-
PROCESSOR 1702
906
VLE
138

FIG. 17

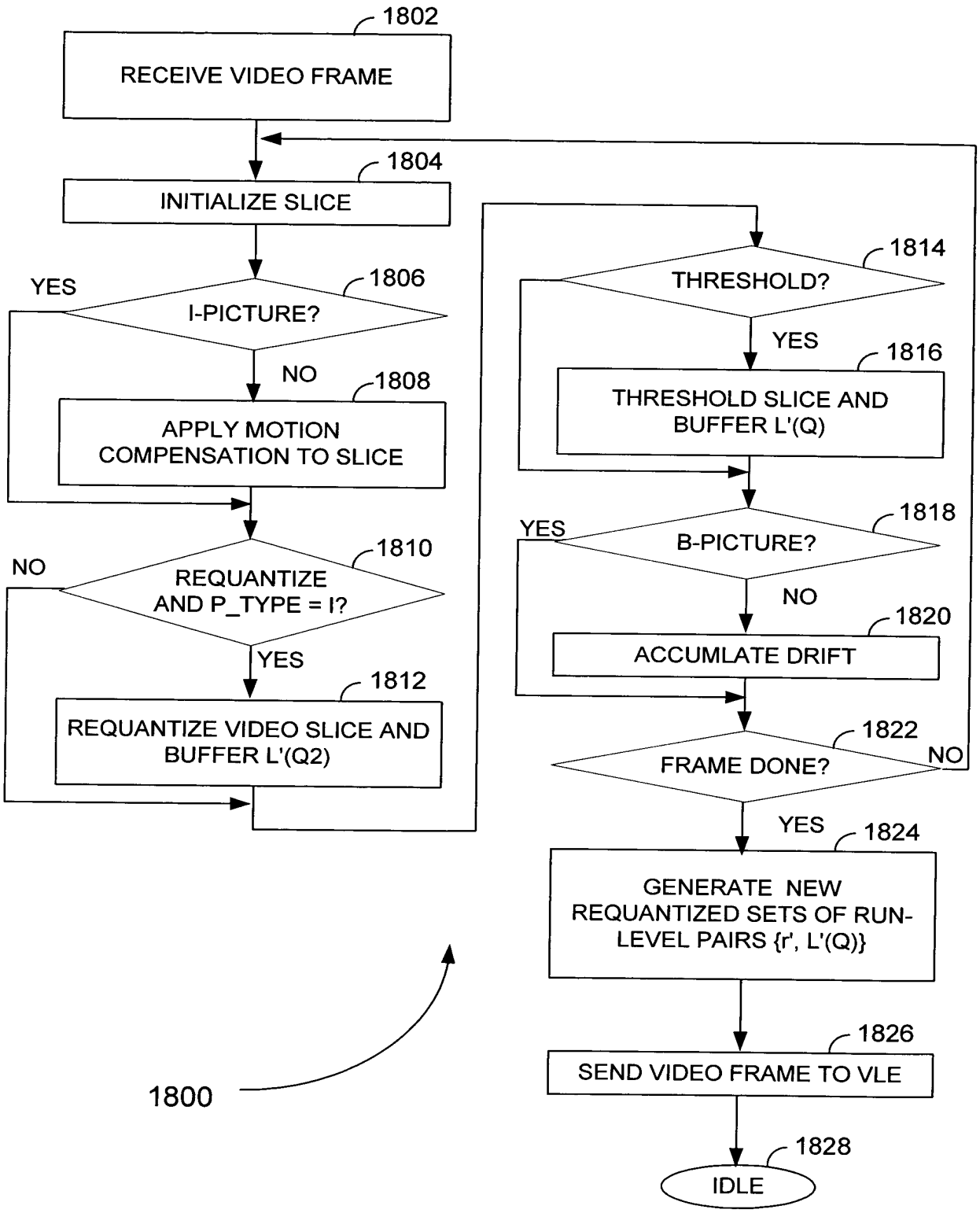

1802
RECEIVE VIDEO FRAME
1804
INITIALIZE SLICE
1806
I-PICTURE?
YES
NO
1808
APPLY MOTION COMPENSATION TO SLICE
1810
REQUANTIZE AND P_TYPE = I?
NO
YES
1812
REQUANTIZE VIDEO SLICE AND BUFFER L'(Q2)
1814
THRESHOLD?
YES
1816
THRESHOLD SLICE AND BUFFER L'(Q)
1818
B-PICTURE?
YES
NO
1820
ACCUMLATE DRIFT
1822
FRAME DONE?
NO
YES
1824
GENERATE NEW REQUANTIZED SETS OF RUN-LEVEL PAIRS {r', L'(Q)}
1826
SEND VIDEO FRAME TO VLE
1828
IDLE
1800

FIG. 18

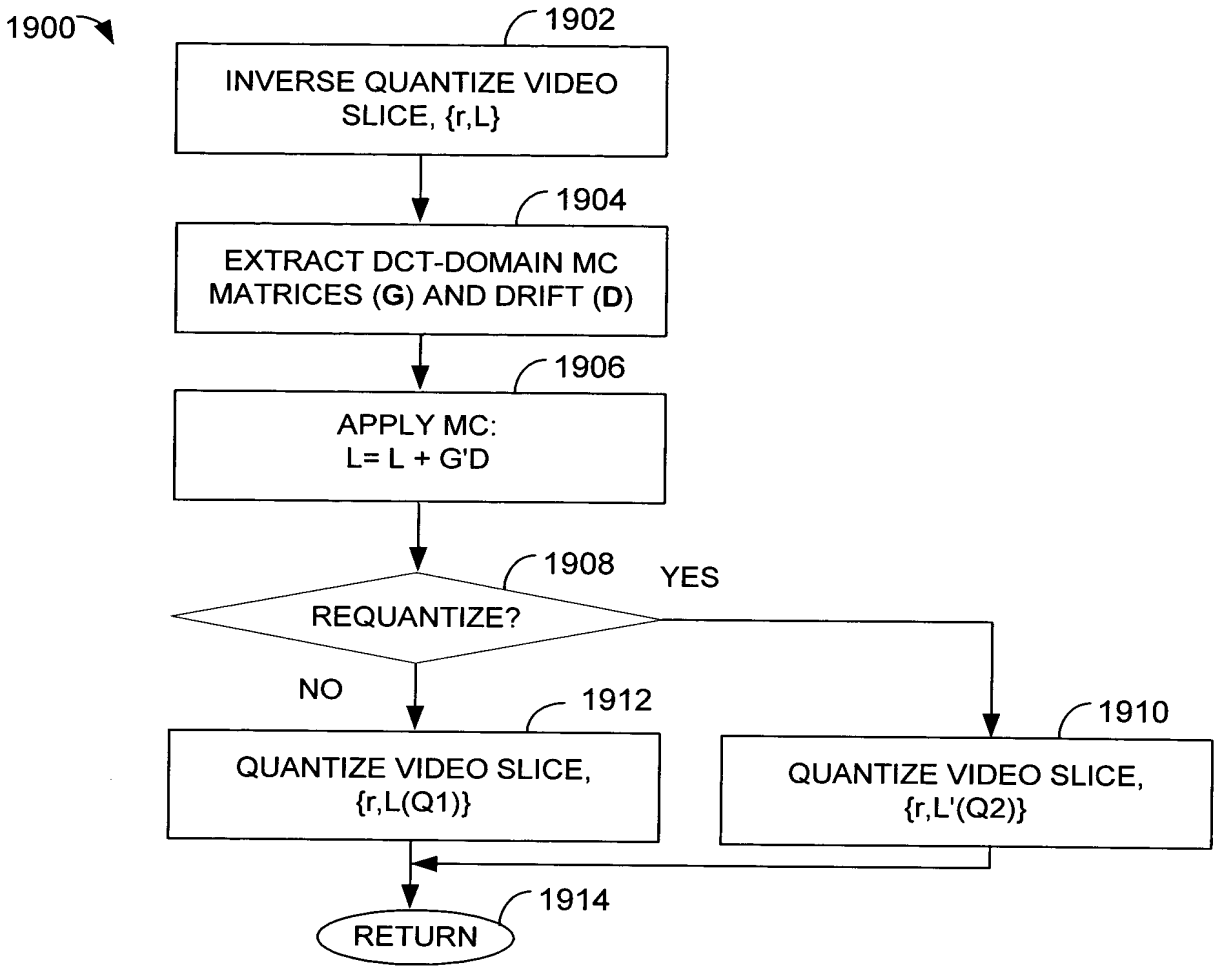

1900
1902
INVERSE QUANTIZE VIDEO SLICE, {r,L}
1904
EXTRACT DCT-DOMAIN MC MATRICES (G) AND DRIFT (D)
1906
APPLY MC: L= L + G'D
1908
REQUANTIZE?
YES
NO
1912
QUANTIZE VIDEO SLICE, {r,L(Q1)}
1910
QUANTIZE VIDEO SLICE, {r,L'(Q2)}
1914
RETURN

FIG. 19

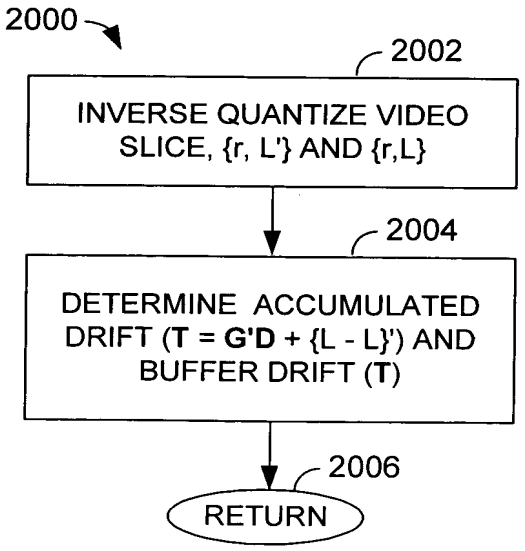

2000
2002
INVERSE QUANTIZE VIDEO SLICE, {r, L'} AND {r,L}
2004
DETERMINE ACCUMULATED DRIFT (T = G'D + {L - L}') AND BUFFER DRIFT (T)
2006
RETURN

FIG. 20